

REMARKS

Claims 1-6, 10-15, 17-18, 21-24, 26-29, 31-32, 35-49, 51-52, 55-58, 60-61, 63, and 66-75 have been amended. Claims 1-75 remain pending in the application. Reconsideration is respectfully requested in light of the following remarks.

Section 101 Rejection:

The Office Action rejected claims 23-25, 37-47, 57-59 and 68-75 under 35 U.S.C. § 101 as allegedly being directed to non-statutory subject matter.

In regard to claims 23-25 and 57-59, the Office Action rejected these claims under 35 U.S.C. § 101, alleging that the claimed inventions are directed to non-statutory subject matter. The Office Action asserts that “these means can be software means...therefore, a system comprising these means is a software system.” Applicants traverse this rejection for at least the following reasons. The elements of 23-25 and 57-59 are all expressed as means for performing a specified function. Applicants remind the Examiner that under 35 U.S.C. § 112, paragraph 6 (emphasis added):

An element in a claim for a combination may be expressed as a means or step for performing a specified function without the recital of underlying structure, material, or acts in support thereof, and such claim shall be construed to cover the corresponding structure, material, or acts described in the specification and equivalents thereof.

Thus, by statutory definition, a means claim specifically includes structure, material, or acts in support thereof and cannot be construed as software *per se*. According to the section of the MPEP on Patentable Subject Matter Eligibility, MPEP 2106.II.C, “Where means plus function language is used to define the characteristics of a machine or manufacture invention, such language must be interpreted to read on only the structures or materials disclosed in the specification and “equivalents thereof” that correspond to the recited function. Two *en banc* decisions of the Federal Circuit have made clear that the USPTO is to interpret means plus function language according to 35 U.S.C. § 112, sixth paragraph. *In re Donaldson*, 16 F.3d 1189, 1193, 29 USPQ2d 1845, 1848 (Fed. Cir.

1994) (en banc); *In re Alappat*, 33 F.3d 1526, 1540, 31 USPQ2d 1545, 1554 (Fed. Cir. 1994) (en banc).” Thus, this rejection is in direct conflict with the MPEP section on Patentable Subject Matter Eligibility. Therefore, for at least the reasons presented above, the § 101 rejection of claims 23-25 and 57-59 is improper and removal thereof is respectfully requested.

In regard to claims 37-47 and 68-75, the Office Action rejected these claims under 35 U.S.C. § 101, alleging that the claimed inventions are directed to non-statutory subject matter, and that “a computer-accessible medium, as defined in the specification to be a transmission medium or signals or links...is non-statutory subject matter.” Applicants traverse this rejection. However, to expedite prosecution, claims 37-47 and 68-75 have been amended to recite “a computer-accessible storage medium.”

Thus, Applicants respectfully request removal of the § 101 rejection of claims 37-47 and 68-75.

Section 112, Second Paragraph, Rejection:

The Office Action rejected claims 1-75 under 35 U.S.C. § 112, second paragraph as indefinite in that the claims recite “acronyms such as API...”, alleging that “these are vague acronyms and uncommon names that must be clearly spelled out in the claims.” Applicants traverse this rejection. However, to expedite prosecution, Applicants have amended relevant ones of claims 1-75 to clarify or replace various acronyms or terms of art used in the claims.

Thus, Applicants respectfully request removal of the §112 rejection of claims 1-75.

Section 103(a) Rejections:

The Office Action rejected claims 1-5, 8, 9, 12-17, 23, 26-31, 34, 37-42, 45, 48-51, 54, 57, 60-62, 65, 68-70 and 73 under 35 U.S.C. § 103(a) as being unpatentable over Joseph (“Developer’s Introduction to JAX-RPC, part 1”) in view of Russell, et al. (WO 03/046757) (hereinafter “Russell”). Applicants traverse these rejections for at least the following reasons.

In regard to claim 1, the Examiner has failed to establish that the cited art discloses, “a client comprising a client Web services stack that supports both a markup language protocol and a binary encoding protocol; and a server comprising a server Web services stack that supports both the markup language protocol and the binary encoding protocol; communicate with the client Web services stack according to the markup language protocol.” The Examiner relies on Joseph to teach the above limitations, but leaves “binary” off of “binary encoding protocol”, asserting that Joseph teaches both a “markup language protocol” and an “encoding protocol.” The Examiner relies on Joseph, FIG. 1, asserting “client and server with service stacks”, and Joseph, table 2 on p. 8, asserting “XML literal (markup language encoding protocol) or other encoded protocol documents.” Table 2 is in a section titled “Mapping concrete WSDL types to Java types.” Referring to Joseph, p. 3, this section appears under the heading “JAX-RPC’s type-mapping system: An introduction.” The paragraph under this heading reads:

One of the most important achievements of the JAX-RPC specification is its definition of a standard for mapping a WSDL document (which represents a service description) to its Java representation (service endpoints, stubs, ties, and Java types), and vice versa...Let’s drill down to the details to see how JAX-RPC is trying to eliminate these problems by defining a common programming model.

Below that on p. 3, in regard to the section “Mapping concrete WSDL types to Java types” in which table 2 appears, Joseph states that the section addresses “the mapping of concrete WSDL definitions (ports, bindings and services) to Java classes.” As stated at page 2, line 5 of Applicants’ specification, “The Web Services Description Language (WSDL) is an XML-based language.”

Thus, this section, which addresses mapping concrete WSDL definitions to Java classes, appears to have little or nothing to do with the subject matter as recited in claim 1. Furthermore, table 2 specifically addresses “SOAP message styles”; col. 1 of table 2 specifically addresses “SOAP message body format”, and col. 2 of table 2 specifically addresses parameter encoding in SOAP messages. Thus, this table is clearly directed at SOAP messages, and has nothing to do with the same client and the same server both supporting both a markup language protocol and a binary encoding protocol.

In further regard to claim 1, the Examiner has failed to establish that the cited art discloses, “wherein the server Web services stack is configured to communicate with the client Web services stack according to the markup language protocol.” The Examiner cites Joseph, table 2 on page 8, asserting “XML markup language protocol.” The table only states that in SOAP, a parameter encoding mode of “literal” uses XML schema for each parameter encoding. Contrary to the Examiner’s assertion, the table does not teach communicating according to a markup language protocol, let alone a server Web services stack communicating with a client Web services stack according to a markup language protocol, as recited in claim 1.

In further regard to claim 1, the Examiner has failed to establish that the cited art discloses, “wherein the client Web services stack and the server Web services stack each support the markup language protocol and the binary encoding protocol with a single API (application programming interface).” The Examiner refers to Joseph, FIG. 1, asserting “single API between client and JAX-RPC stub or JAX-RPC ties.” FIG. 1 clearly shows “JAX-RPC Runtime (**APIs**)” on both client and server. **APIs** is clearly plural. Thus, contrary to the Examiner’s assertion, this Figure does not show a “single API between client and JAX-RPC stub or JAX-RPC ties.”

In further regard to claim 1, the Examiner has failed to establish that the cited art discloses “wherein the server Web services stack is configured to: communicate with the client Web services stack according to the markup language protocol; and dynamically

switch to communicate with the client Web services stack according to the binary encoding protocol.”

The Examiner relies on Russell to disclose the above limitations, citing Russell, p. 3, summary paragraph 1, and asserting “determining whether the binary encoding code book (of a binary encoding protocol) is stored on the device, use the code book to use the binary encode-decode protocol to recover the XML document.” **The Examiner leaves out an important part of this citation**, which further states: “requesting the code book from the data server where the code book is not stored on the wireless mobile communication device, receiving the code book from the data server.” Thus, Russell specifically teaches a method for processing documents on a wireless mobile communications device that involves receiving a document (transcoded from an XML document using a code book) from a server, determining that the code book is not on the device, requesting the code book from the server, receiving the code book from the server, and transcoding the received document using the code book to recover the XML document. Russell’s system is clearly nothing at all like what is recited in Applicants’ claim 1, which in contrast to Russell’s teachings, recites a server Web services stack configured to communicate with a client Web services stack according to a markup language protocol; and dynamically switch to communicate with the client Web services stack according to a binary encoding protocol. Russell’s method of transcoding XML documents with code books on a server and downloading code books to wireless devices to decode received transcoded documents to thus reproduce XML documents on the devices **has absolutely nothing to do with** dynamically switching from communicating between a server and a client according to a markup language protocol to communicating between the server and the client according to a binary encoding protocol.

Combining Russell’s method with Joseph’s teachings, even if possible, would not produce anything like what is recited in claim 1. Such a combination would appear to instead produce a JAX-RPC system in which a device could receive a transcoded document from a server, request a code book for the transcoded document, receive the code book, and decode the transcoded document to produce an XML document.

Furthermore, Russell's method has nothing to do with "web services communications", and Applicants fail to see how Russell's method, if combined with Joseph's teachings, would or could achieve the objective of "in order to choose a most suitable protocol for implementing web services communications", as asserted by the Examiner. Moreover, the Examiner's reason to combine the references is merely conclusory. Thus, the Examiner has failed to provide a proper *prima facie* reason to combine the references.

Thus, for at least the reasons presented above, the rejection of claim 1 is unsupported by the cited art and removal thereof is respectfully requested. Similar remarks apply equally to claims 12, 23, 26, and 37.

In regard to claim 48, similar remarks regarding the Joseph reference apply equally to the Examiner's assertions using Joseph in regards to "communicate with other systems using either a binary encoding protocol or a markup language protocol using a single API (application programming interface)" as recited in claim 48.

In further regard to claim 48, contrary to the Examiner's assertion, the cited art fails to disclose "a Web services stack configured to: negotiate with another system to determine if the other system supports the binary encoding protocol; if the other system supports the binary encoding protocol, communicate with the other system according to the binary encoding protocol" The Examiner asserts Russell teaches these limitations at p. 3, the Summary, paragraph 1, and asserts "determining whether the binary encoding code book is stored on the device, use the code book to recover the XML document." Contrary to the Examiner's assertions, Russell's teachings, which were thoroughly discussed above regarding claim 1, do not disclose anything like these limitations as recited in claim 48. Russell's systems do not "negotiate with another system to determine if the other system supports the binary encoding protocol", Russell's "device" is not determined to "support the binary encoding protocol", and Russell's server does not "communicate with the other system according to the binary encoding protocol". In

contrast, Russell's server simply sends a transcoded document to a device. The device requests a code book from the server, and uses the code book to decode the device. There is no negotiation, the device is not "determined to support the binary encoding protocol" prior to communicating with the device, and there is no communication between the devices according to a binary encoding protocol described.

In further regard to claim 48, the cited art does not teach "if the other system does not support the binary encoding protocol, communicate with the other system according to the markup language protocol." The Examiner relies on Joseph, citing Joseph, table 2, asserting "XML encoding protocol is required as a basis encoding protocol for a JAX-RPC web service stack." As noted above in regard to claim 1, table 2 is directed at SOAP, not XML, and the table appears in a section addressing mapping WSDL concrete types to Java types. Furthermore, the table only states that in SOAP, a parameter encoding mode of "literal" uses XML schema for each parameter encoding. Contrary to the Examiner's assertion, the table does not teach communicating according to a markup language protocol as recited in claim 48.

In further regard to claim 48, combining Russell's method with Joseph's teachings, if possible, would not produce anything like what is recited in claim 48 when viewed as a whole. Claim 48, when viewed as a whole, recites "a Web services stack configured to: communicate with other systems using either a binary encoding protocol or a markup language protocol using a single API; negotiate with another system to determine if the other system supports the binary encoding protocol; if the other system supports the binary encoding protocol, communicate with the other system according to the binary encoding protocol; and if the other system does not support the binary encoding protocol, communicate with the other system according to the markup language protocol." No combination of the cited references would produce the subject matter as recited in the claim.

A combination of Russell with Joseph, instead of producing the subject matter as recited in claim 48, would appear to simply produce a JAX-RPC system in which a

device could receive a transcoded document from a server, request a code book for the transcoded document, receive the code book, and decode the transcoded document to produce an XML document. Such a combination would clearly produce nothing like what is recited in claim 48.

Furthermore, Russell's method has nothing to do with "web services communications", and Applicants fail to see how Russell's method, if combined with Joseph's teachings, would or could achieve the objective of "in order to choose a most suitable protocol for implementing web services communications", as asserted by the Examiner. Moreover, the Examiner's reason to combine the references is merely conclusory. Thus, the Examiner has failed to provide a proper *prima facie* reason to combine the references.

Thus, for at least the reasons presented above, the rejection of claim 48 is unsupported by the cited art and removal thereof is respectfully requested. Similar remarks apply equally to claims 57, 60, and 68.

The Office Action rejected claims 6, 10, 11, 18, 21, 22, 24, 32, 35, 36, 43, 46, 47, 50, 55, 56, 58, 63, 66, 67, 71, 74 and 75 under 35 U.S.C. § 103(a) as being unpatentable over Joseph and Russell in view of Krill ("Binary Encodings Key to Proposal"). Applicants respectfully traverse this rejection for at least the reasons given above in regard to the independent claims. Krill does not include any teachings that overcome any of the above-noted deficiencies of the cited art in regard to the independent claims.

The Office Action rejected claims 7, 19, 33, 44, 53, 59, 64 and 72 under 35 U.S.C. § 103(a) as being unpatentable over Joseph and Russell in view of ASN.1 (Introduction to ASN.1, IDS). Applicants respectfully traverse this rejection for at least the reasons given above in regard to the independent claims. ASN.1 does not include any teachings that overcome any of the above-noted deficiencies of the cited art in regard to the independent claims.

Applicants also assert that the rejection of numerous ones of the dependent claims is further unsupported by the cited art. However, since the rejections have been shown to be unsupported for the independent claims, a further discussion of the dependent claims is not necessary at this time.

CONCLUSION

Applicants submit the application is in condition for allowance, and an early notice to that effect is respectfully requested.

If any fees are due, the Commissioner is authorized to charge said fees to Meyertons, Hood, Kivlin, Kowert, & Goetzel, P.C. Deposit Account No. 501505/5681-74200/RCK.

Respectfully submitted,

/Robert C. Kowert/
Robert C. Kowert, Reg. #39,255
Attorney for Applicants

Meyertons, Hood, Kivlin, Kowert, & Goetzel, P.C.
P.O. Box 398
Austin, TX 78767-0398
Phone: (512) 853-8850

Date: June 9, 2009